

Contents

1 Routine/Function Prologues	2
1.1 Fortran: Module Interface grid_spmdMod.F90 (Source File: grid_spmdMod.F90)	2
1.1.1 allocate_gdd (Source File: grid_spmdMod.F90)	2
1.1.2 grid_spmd_init (Source File: grid_spmdMod.F90)	2

1 Routine/Function Prologues

1.1 Fortran: Module Interface grid_spmdMod.F90 (Source File: grid_spmdMod.F90)

This module computes domain decomposition on the grid domain

REVISION HISTORY:

14 Nov 2002 Sujay Kumar Initial Specification

INTERFACE:

```
module grid_spmdMod
```

USES:

```
use spmdMod
```

ARGUMENTS:

```
integer, allocatable :: gdi(:), gdisp(:)
integer, allocatable :: g2di(:), g2disp(:)
```

1.1.1 allocate_gdd (Source File: grid_spmdMod.F90)

Allocates memory for arrays that contain domain decomposition information

INTERFACE:

```
subroutine allocate_gdd()
```

DESCRIPTION:

Allocates memory for arrays that contain domain decomposition information

```
allocate(gdi(0:npes-1))
allocate(gdisp(0:npes-1))
allocate(g2di(0:npes-1))
allocate(g2disp(0:npes-1))
```

1.1.2 grid_spmd_init (Source File: grid_spmdMod.F90)

Computes domain decomposition based on the number of processors

INTERFACE:

```
subroutine grid_spmd_init(tile,nch,nmif,ngrid)
```

USES:

```
use tile_module
use tile_spmdMod, only : displs
!INPUT ARGUMENTS:
integer :: nch, ngrid, nmif
!OUTPUT ARGUMENTS:
type(tiledec)::tile(nch)
```

CONTENTS:

```
gdisp(0) = 0
do p=1, npes-1
    gdisp(p) = tile(displs(p))%index
enddo
do p = 0 , npes-2
    gdi(p) = gdisp(p+1)-gdisp(p)
enddo
gdi(npes-1) = ngrid - gdisp(npes-1)
do p=0, npes-1
    g2di(p) = gdi(p)*nmif
enddo
g2disp(0) = 0
do p=1, npes-1
    g2disp(p) = g2disp(p-1)+g2di(p-1)
enddo
```